Final Technical Report for $\mathcal{MLC}$++
A Machine Learning Library in C++

Nils J. Nilsson
Stanford University

May 22, 1995

This report details the research and development work done on $\mathcal{MLC}$++ under ONR grant N00014-94-1-0448.

1 Overview of $\mathcal{MLC}$++

$\mathcal{MLC}$++ is a Machine Learning library of C++ classes. General information about the library can be obtained through the World Wide Web at URL

`http://robotics.stanford.edu:/users/ronnyk/mlc.html` .

The current implementation supports supervised learning of concepts using decision trees, decision graphs, nearest-neighbor (instance-based), and probabilities (Naive-Bayes). Algorithms for feature subset selection and discretization can work with any of the induction algorithms.

$\mathcal{MLC}$++ object code for Sun is available through the World Wide Web. Over 150 different sites have copied the $\mathcal{MLC}$++ kit, and machine learning research in the robotics lab at Stanford is enhanced through the use of the library. All the algorithms in Ron Kohavi's dissertation, for example, are implemented in $\mathcal{MLC}$++.

2 Summary of Results

As detailed in the statement of work for the grant, three main projects were proposed:

1. Search algorithms.
2. General Logic Diagrams (GLDs).
3. Data manipulation routines.

We now describe the specific work done and the results obtained.

2.1 Search algorithms

Hill climbing and best-first search were implemented as general search algorithms. Attempts to use the search techniques for finding small decision trees, as originally envisioned, did not result in significant performance improvements; however, the algorithms were then used for a different purpose, feature subset selection, and important research results were obtained.

In John, Kohavi & Pfleger (1994), the *wrapper approach* to feature subset selection was proposed. The problem of feature subset selection is that of finding features that are relevant to the supervised task at hand. Feature subset selection has been studied for many years in statistics, pattern recognition, and machine learning; however, most suggestions were based on a *filter approach* where the data alone determined what features are important, thus ignoring the induction algorithm. The proposed approach uses the induction algorithm as a black box and testing its performance on different feature subsets to determine the best set of features for future predictions. In Kohavi (1994*a*), the problem was generalized and abstracted into a search with probabilistic estimates. *Best-first-search* was used and was shown to be superior to hill-climbing.

The work on feature subset selection concentrated on decision-trees as the underlying hypothesis space; ID3 (Quinlan 1986) and C4.5 (Quinlan 1993) were used as the underlying induction algorithms. An observation was made that very few features were actually chosen by the algorithm and that most trees were complete, *i.e.*, they tested all the features. This suggested that much of the inductive power comes from finding a relevant set of features, not from the actual tree-structure that was used. Testing the conjecture using $\mathcal{MLC}$++ was extremely easy; the same day, we had results showing that, indeed, for discrete datasets, performance of decision tables on features selected by the wrapper approach was comparable to that of the best induction algorithms. The work was reported in Kohavi & Frasca (1994) and a more systematic study with a better understanding of the underlying phenomena was reported in Kohavi (1995*a*). We believe that this surprising result would never have been discovered without the power of $\mathcal{MLC}$++. Testing the conjecture without $\mathcal{MLC}$++ would have required a long time, and it probably would never have been done.

Recent work on feature subset selection using dynamic operators for the search space and the use of other induction algorithms was reported in Kohavi & Sommerfield (1995) together with a discussion on overfitting in feature subset space.

Another use for the wrapper approach is that of *parameter tuning.* Given an algorithm with different possible settings, how can one find a good setting for the task. Kohavi & John (1995) reported significant improvements to C4.5 Quinlan (1993) when these parameters were tuned automatically using the wrapper approach.

2.2 General Logic Diagrams

General Logic Diagrams, or GLDs, were originally proposed by Michalski (Michalski 1978). The diagrams allow viewing multi-dimensional discrete spaces and can help researchers gain insight to the induced concept by inspecting it.

GLDs were implemented in $\mathcal{MLC}$++ and were used for illustrative purposes in (Kohavi 1994*b*).

2.3 Data manipulation

Data conversions to local and binary encodings were implemented. Three algorithms for discretization of continuous features were implemented: uniform binning, the 1R discretization proposed in Holte (1993), and the entropy-based discretization proposed in Fayyad & Irani (1993) and Catlett (1991). The methods were compared in Dougherty, Kohavi & Sahami (1995). The Naive-Bayes algorithm (Langley, Iba & Thompson 1992) was shown to dramatically improve in accuracy after discretization.

2.4 Related Projects

The ONR grant was acknowledge in papers that were not directly related to the grant, but which nonetheless indirectly profited from the supported work (Kohavi 1995*b*, Kohavi & Li 1995, Kohavi, John, Long, Manley & Pfleger 1994).

3 Summary

$\mathcal{MLC}$++ has been extremely helpful in our research and is currently helping other researchers in comparing different algorithms for given datasets. Work on the library is continuing in an effort to improve the quality and enlarge the number of useful tools we can provide.

The main research contribution was the work on feature subset selection. The proposed wrapper approach was very successful and was already used by other researchers (Langley & Sage 1994, Aha & Bankert 1994*b*, Aha & Bankert 1994*a*, Mladenić 1995). The implementation of the different discretization algorithms has led to a better understanding of the methods. In some cases (most notably, Naive-Bayes), performance using the discretized data is significantly better, surpassing that of the best known algorithms for many datasets. The implementation of general logic diagrams provides researchers with another tool for viewing data.

References

Aha, D. W. & Bankert, R. L. (1994*a*), A comparative evaluation of sequential feature selection algorithms, *in* "Proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics", pp. 1–7.

Aha, D. W. & Bankert, R. L. (1994*b*), Feature selection for case-based classification of cloud types: An empirical comparison, *in* "Working Notes of the AAAI-94 Workshop on Case-Based Reasoning", pp. 106–112.

Catlett, J. (1991), On changing continuous attributes into ordered discrete attributes, *in* Y. Kodratoff, ed., "Proceedings of the European Working Session on Learning", Berlin, Germany: Springer-Verlag, pp. 164–178.

Dougherty, J., Kohavi, R. & Sahami, M. (1995), Supervised and unsupervised discretization of continuous features, *in* "Machine Learning: Proceedings of the Twelfth International Conference". Available by anonymous ftp from `starry.Stanford.EDU:pub/ronnyk/disc.ps`.

Fayyad, U. M. & Irani, K. B. (1993), Multi-interval discretization of continuous-valued attributes for classification learning, *in* "Proceedings of the 13th International Joint Conference on Artificial Intelligence", Morgan Kaufmann, pp. 1022–1027.

Holte, R. C. (1993), "Very simple classification rules perform well on most commonly used datasets", *Machine Learning* **11**, 63–90.

John, G., Kohavi, R. & Pfleger, K. (1994), Irrelevant features and the subset selection problem, *in* "Machine Learning: Proceedings of the Eleventh International Conference", Morgan Kaufmann, pp. 121–129. Available by anonymous ftp from: `starry.Stanford.EDU:pub/ronnyk/ml94.ps`.

Kohavi, R. (1994*a*), Feature subset selection as search with probabilistic estimates, *in* "AAAI Fall Symposium on Relevance", pp. 122–126. Available by anonymous ftp from: `starry.Stanford.EDU:pub/ronnyk/aaaiSymposium94.ps`.

Kohavi, R. (1994*b*), A third dimension to rough sets, *in* "Third International Workshop on Rough Sets and Soft Computing", pp. 244–251. Available by anonymous ftp from: `starry.Stanford.EDU:pub/ronnyk/roughOODG.ps`.

Kohavi, R. (1995*a*), The power of decision tables, *in* N. Lavrac & S. Wrobel, eds, "Machine Learning: ECML-95 (Proc. European Conf. on Machine Learning, 1995)", Lecture Notes in Artificial Intelligence 914, Springer Verlag, Berlin, Heidelberg, New York, pp. 174 – 189. Available by anonymous ftp from `starry.Stanford.EDU:pub/ronnyk/tables.ps`.

Kohavi, R. (1995*b*), A study of cross-validation and bootstrap for accuracy estimation and model selection, *in* "Proceedings of the 14th International Joint Conference on Artificial Intelligence". Available by anonymous ftp from `starry.Stanford.EDU:pub/ronnyk/accEst.ps`.

Kohavi, R. & Frasca, B. (1994), Useful feature subsets and rough set reducts, *in* "Third International Workshop on Rough Sets and Soft Computing", pp. 310–317. Available by anonymous ftp from: `starry.Stanford.EDU:pub/ronnyk/rough.ps`.

Kohavi, R. & John, G. (1995), Automatic parameter selection by minimizing estimated error, *in* "Machine Learning: Proceedings of the Twelfth International Conference".

Kohavi, R. & Li, C.-H. (1995), Oblivious decision trees, graphs, and top-down pruning, *in* "Proceedings of the 14th International Joint Conference on Artificial Intelligence". Available by anonymous ftp from `starry.Stanford.EDU:pub/ronnyk/eodg.ps`.

Kohavi, R. & Sommerfield, D. (1995), Feature subset selection using the wrapper model: Overfitting and dynamic search space topology, *in* "The First International Conference on Knowledge Discovery and Data Mining". Available by anonymous ftp from `starry.Stanford.EDU:pub/ronnyk/fssWrapper.ps`.

Kohavi, R., John, G., Long, R., Manley, D. & Pfleger, K. (1994), MLC++: A machine learning library in C++, *in* "Tools with Artificial Intelligence", IEEE Computer Society Press, pp. 740–743. Available by anonymous ftp from: `starry.Stanford.EDU:pub/ronnyk/mlc/toolsmlc.ps`.

Langley, P. & Sage, S. (1994), Induction of selective bayesian classifiers, *in* "Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence", Morgan Kaufmann, Seattle, WA, pp. 399–406.

Langley, P., Iba, W. & Thompson, K. (1992), An analysis of bayesian classifiers, *in* "Proceedings of the tenth national conference on artificial intelligence", AAAI Press and MIT Press, pp. 223–228.

Michalski, R. S. (1978), A planar geometric model for representing multidimensional discrete spaces and multiple-valued logic functions, Technical Report UIUCDCS-R-78-897, University of Illinois at Urbaba-Champaign.

Mladenić, D. (1995), Automated model selection, *in* "ECML workshop on Knowledge Level Modeling and Machine Learning".

Quinlan, J. R. (1986), "Induction of decision trees", *Machine Learning* **1**, 81–106. Reprinted in Shavlik and Dietterich (eds.) Readings in Machine Learning.

Quinlan, J. R. (1993), *C4.5: Programs for Machine Learning*, Morgan Kaufmann, Los Altos, California.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
950523

3. REPORT TYPE AND DATES COVERED
final 940201 - 950131

4. TITLE AND SUBTITLE
MLC++ - A Machine Learning Library in C++

5. FUNDING NUMBERS
N00014-94-1-0448

6. AUTHOR(S)
Nils J. Nilsson

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Dept. of Computer Science
Stanford University
Stanford, CA 94305

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Naval Research
800 N. Quincy St.
Arlington, VA 22217-5660

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
approved for public release: distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

see attached report

14. SUBJECT TERMS

15. NUMBER OF PAGES
5

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT
UL

18. SECURITY CLASSIFICATION OF THIS PAGE
UL

19. SECURITY CLASSIFICATION OF ABSTRACT
UL

20. LIMITATION OF ABSTRACT
UL